DETAILED ACTION

Claims 11-13 and 15-20 remain for examination.

Claim Rejections - 35 USC § 102

1. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

2. Claims 11-13 and 15-20 are rejected under 35 U.S.C. 102(b) as being anticipated by *Rotenberg ("AR-SMT: A Microarchitectural Approach to Fault Tolerance in Microprocessors")* herein referred to as Rotenberg.

Claim 11

Rotenberg teaches: **A method comprising: dividing a dynamic sequential program into multiple epochs comprising a first epoch instance and a second epoch instance** (Figure 2, A-Stream and R-Stream, Section 1.2 *Note: The R-stream lags behind the A-stream and is therefore a trailing thread*)**, wherein each epoch includes two or more instructions** (Section 1.2 *Note: The streams each have multiple instructions*)**; in a redundant multi-threading (RMT) system having leading and trailing threads** (Figure 2, A-Stream and R-Stream, Section 1.2)**, redundantly executing in parallel the first epoch instance and second epoch instance for each epoch as the leading and trailing threads, respectively** (Figure 2, A-Stream and R-Stream, Section 1.2)**; for the executed first epoch instance and second epoch instance, saving store results of the**

first epoch instance and the second epoch instance as speculative stores to memory, the speculative stores being exposed (Figure 2, Delay Buffer, Section 1.2 Paragraph 2, "As the R-stream is fetched and executed, it's committed results are compared to those in the Delay Buffer" *Note: also see Paragraph 1 of section 1.2 which notes that results of the A-Stream are committed to memory along with the results of the R-stream, thus both are saved)*; **comparing the saved exposed stores** (Section 1.2, Paragraph 2, *Note: The results of the A-Stream are stored in the Delay Buffer for comparison*); **and when the exposed stores match, committing a single set of the exposed stores to an architectural memory state corresponding to the dynamic sequential program** (Section 1.2, Paragraph 2 *Note: If the comparison fails then a fault is detected and the results would not be committed as the actual result of the instruction given that the result is a known fault*).

Claim 12

Rotenberg teaches: **The method of claim 11, wherein the speculative stores are from a re-order buffer** (Section 1.2 *Note: As the status of the results could still be a fault they are inherently speculative*).

Claim 13

Rotenberg teaches: **The method of claim 12, wherein the two or more instructions executed in response to the execution of the first and second epoch instances are buffered prior to epoch execution completion** (Figure 2, Delay Buffer).

Claim 15

Rotenberg teaches: **The method of claim 11, wherein the memory is L1 cache memory** (Figure 4).

Claims 16-18 and 20

Claims 16-18 and 20 contain the same limitations as claims 11-13, 15, and 19 and are rejected for the same reasons set forth in connection with the rejections of claims 11-13, 15, and 19.

Claim 19

Rotenberg teaches: **The method of claim 16, further comprising committing the store results of a first epoch instance or second epoch instance to a sequential architectural state of the computation in response to the first epoch instance or second epoch instance becoming an oldest epoch.** (Section 1.2, Figure 2 *Note: The instructions are committed in order from the delay buffer, a FIFO queue, which means as the oldest will be the first in, it will also be the first out and thus the first committed).*

3. Examiner believes the above rejection is sufficient for anticipating the claimed invention. However, in the event that Applicant disagrees, Examiner presents the below rejection as further evidence regarding the lack of novelty in the claims.

4. Claims 11-13 and 15-20 are rejected under 35 U.S.C. 102(b) as being anticipated by *Reinhardt et al. ("Transient Fault Detection via Simultaneous Multithreading")* herein referred to as Reinhardt.

Claim 11

Reinhardt teaches: **A method comprising: dividing a dynamic sequential program into multiple epochs comprising a first epoch instance and a second epoch instance** (Figure 3, Thread 0 and Thread 1, Section 3.1 *Note: The redundant thread may be time redundant and is therefore a trailing thread*)**, wherein each epoch includes two or more instructions** (Section 3.1 *Note: The threads each have multiple instructions*)**; in a redundant multi-threading (RMT) system having leading and trailing threads** (Figure 3, Thread 0 and Thread 1)**, redundantly executing in parallel the first epoch instance and second epoch instance for each epoch as the leading and trailing threads, respectively** (Figure 3, Thread 0 and Thread 1, Section 3.1)**; for the executed first epoch instance and second epoch instance, saving store results of the first epoch instance and the second epoch instance as speculative stores to memory, the speculative stores being exposed** (Section 3.2, Paragraph 2, Paragraph 7, "register writeback comparison…" *Note: The results are stored in a register check buffer or alternatively in a store buffer, depending on the inclusion of the registers in the sphere of replication*)**; comparing the saved exposed stores** (Section 3.2, Paragraph 7, "register writeback comparison…" *Note: The values are compared and if they match then the value is committed*)**; and when the exposed stores match, committing a single set of the exposed stores to an architectural memory state corresponding to the dynamic**

sequential program (Section 3.2, Paragraph 7, "register writeback comparison..." *Note: The values are compared and if they match then the value is committed*).

Claim 12

Reinhardt teaches: **The method of claim 11, wherein the speculative stores are from a re-order buffer** (Section 3.2, Paragraph 7, "register writeback comparison..." *Note: As the instructions have yet to be committed they are still speculative*).

Claim 13

Reinhardt teaches: **The method of claim 12, wherein the two or more instructions executed in response to the execution of the first and second epoch instances are buffered prior to epoch execution completion** (Section 3.2, Paragraph 7, "register writeback comparison...").

Claim 15

Reinhardt teaches: **The method of claim 11, wherein the memory is L1 cache memory** (Figure 2).

Claims 16-18 and 20

Claims 16-18 and 20 contain the same limitations as claims 11-13, 15, and 19 and are rejected for the same reasons set forth in connection with the rejections of claims 11-13, 15, and 19.

Claim 19

Reinhardt teaches: **The method of claim 16, further comprising committing the store results of a first epoch instance or second epoch instance to a sequential architectural state of the computation in response to the first epoch instance or second epoch instance becoming an oldest epoch.** (Section 2.2, "committing results… in program order").

Response to Arguments

5. Applicant's arguments filed 5/21/2010 have been fully considered but they are not persuasive. Applicant argues in substance:

a. In Rotenberg, instructions executed in the "A-stream" have their results committed and placed into a delay buffer. As such, these results are not "speculative." Additionally, the results of the execution of the same instruction in the "R-stream" are never stored. Rather, these results compared to those stored in the delay buffer. As such, Rotenberg does not describe this limitation.

i. Examiner respectfully disagrees. Both results from the A-Stream and R-Stream are stored in memory (the results are committed, thus stored, as per section 1.2). As it is not yet known if either are in error and they must still be checked the results are still speculative by definition - it's not known if they are correct or not. The results from the R-stream as also stored (Section 1.2 "As the R-stream is fetched an executed, its committed results are compared to those in the Delay Buffer" – in order to be committed they must be stored).

b. Second, Rotenberg does not describe "comparing the exposed stores" saved in memory are not compared.

ii. Examiner respectfully disagrees. As the committed results from the R-stream are compared to those in the Delay Buffer, which contains the results of A-stream, then the stores are compared (Section 1.2).

c. Third, Rotenberg does not describe "if the exposed stores match, committing a single set of the exposed stores to an architectural memory state corresponding to the dynamic sequential program." As detailed by Rotenberg, in his system the results of the "A-stream" are committed regardless of when the comparison of the delay buffer contacts and the "R-stream" says. Thus does not describe this limitation.

iii. Examiner respectfully disagrees. As Applicant notes there are two places to which results are saved, the first being a memory and the second being the architectural memory state. While results are committed to memory, they are not finalized until the comparison is done, similar to saving to the architectural memory state as until they are compared they may still have a fault (Section 1.2).

d. First, Reinhardt does not describe "for the executed first epoch instance and second epoch instance, saving store results of the first epoch instance and the second epoch instance as speculative stores to memory, the speculative stores being exposed." In the cited section, Reinhardt describes recording the results of a first instruction instance into a register check buffer. "When the corresponding instruction from the other thread leaves the RUU, the index and value are compared

and, if they match, the register file is updated." However, there is no saving of the result of the second instruction instance.

iv. Examiner respectfully disagrees. Reinhardt shows two ways of dealing with stores, depending on if the registers are or are not included in the sphere of replication (FIG. 3(a) vs 3(b)). Examiner notes that in both events that the results are stored, either in a store buffer, as per 3(a), or in a register check buffer, as per 3(b) (Section 3.2 - "We use an ordered, non-coalescing store buffer shared between the redundant threads to synchronize and verify store values as they retire in program order…").

e. Second, Reinhardt does not describe "comparing the saved exposed stores" as it does not describe the previous limitation.

v. Examiner respectfully disagrees. Specifically the results are compared (Section 3.2 - "…will compare its address and data…" and "…the index and value are compared...").

f. Finally, Reinhardt does not describe "when the exposed stores match, committing a single set of the exposed stores to an architectural memory state corresponding to the dynamic sequential program." Again, the second instance is not saved and then compared, and as such, there is no commit based upon such as saved store.

vi. Examiner respectfully disagrees. In at least one method the stored results are compared and if they match the result is stored in the data cache, thus

committing it to the architectural state (Section 3.2 "On a match, the entry is marked as verified and issued to the data cache").

6. In the interest of compact prosecution, Examiner notes that claims are afforded their broadest reasonable interpretation. It appears, in this case, Applicant may be affording more weight to certain limitations than the claims explicitly require. Specifically with regards to the speculative nature and other limitations. Examiner notes in this case that speculative is interpreted to be that of "conjectural" or "based on conjecture rather than knowledge" and as the results are not known to be correct or false until comparison, they are simply speculative until confirmation.

7. Examiner notes that an interview may help clarify the limitations, specifically in regards to interpretation and either allows the record to be made clear or to result in possible amendments which may facilitate the allowance of the application.

Conclusion

1. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

8. A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event,

however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

9. Examiner respectfully requests, in response to this Office action, support be shown for language added to any original claims on amendment and any new claims. That is, indicate support for newly added claim language by specifically pointing to page(s) and line number(s) in the specification and/or drawing figure(s). This will assist Examiner in prosecuting the application.

10. When responding to this Office Action, Applicant is advised to clearly point out the patentable novelty which he or she thinks the claims present, in view of the state of the art disclosed by the references cited or the objections made. He or she must also show how the amendments avoid such references or objections. See 37 CFR 1.111(c).

11. Any inquiry concerning this communication or earlier communications from the examiner should be directed to William B. Partridge whose telephone number is (571) 270-1402. The examiner can normally be reached on M-F 2:00 - 6:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (571) 272-4162. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/William B Partridge/
Examiner, Art Unit 2183

/EDDIE P CHAN/

Supervisory Patent Examiner, Art Unit 2183